Beschreibung der Leistungsobjekte

“Weiterentwicklung und Betrieb der Software für die Datentreuhänder-Plattform EuroDaT sowie des AML-Anwendungsfalls safeAML”

1. EuroDaT-Plattform

EuroDaT-Plattform ist das Leistungsobjekt, das im Rahmen von Los 1 weiterzuentwickeln und zu betreiben ist.

EuroDaT-Plattform ist die Software, mit der EuroDaT GmbH ihre Dienste als Datentreuhänderin anbietet. Der Quellcode der Software wird als Open Source (unter BSD 3-Clause) entwickelt und auf GitLab maintained.

Die EuroDaT-Plattform wird in der Google Cloud betrieben. Die Google Cloud-Organisation der EuroDaT GmbH ist Teil des Leistungsobjekts.

1.1. Allgemeines

Die Grundidee der EuroDaT-Plattform ist es zu ermöglichen, dass Drittanbieter (oder EuroDaT selbst) Apps (auch Use Cases genannt) anbieten, mit Hilfe derer von Datengebern bereitgestellte Eingabe-Daten verarbeitet und die Ergebnisse den Analysenehmern ausgeliefert werden können. Dabei soll die (durch die Apps definierte) Verarbeitung so erfolgen, dass kein Unberechtigter (insbesondere nicht der Anbieter der App) in die Verarbeitung eingreifen oder Daten exfiltrieren kann. Die Verarbeitung erfolgt in “Transaktionen”, zu deren Beginn die Teilnehmer und ihre Rollen festgelegt und Ressourcen zur Datenverarbeitung bereitgestellt werden, und an deren Ende die bereitgestellten Daten und Ressourcen gelöscht werden.

Teilnehmer sind Unternehmen. Sie können die Rolle “Datendienstleister” (= App-Anbieter) oder im Rahmen einer App “Datengeber” (kann Eingabedaten schreiben), “Workflow-Ausführer” (kann Verarbeitungsschritte = Workflows starten) und/oder “Analysenehmer” (kann Ergebnisse lesen) einnehmen.

Apps bestehen aus einer Liste von Workflows mit passenden Container-Images, einer Schema-Definition für Eingabe-, Intermediär- und Ergebnisdaten, sowie aus allgemeinen Regeln für die Zuweisung von Rollen.

Teilnehmer können Clients registrieren und mit diesen M2M-Interaktionen mit der EuroDaT-API im Rahmen von Apps ausführen, z.B. Transaktion starten/beenden, Workflow starten/beenden, Nachrichten abholen, Eingabedaten schreiben, Ergebnisdaten lesen.

Da Teilnehmer sich bei EuroDaT selbst oder für eine App registrieren können, gibt es eine Web UI, auf der man sich anmelden und Onboarding- oder App-bezogene Verträge schließen kann. (Dafür ist das Onboarding von Zeichnungsberechtigten des Teilnehmer-Unternehmens erforderlich.) Man kann administrative Tätigkeiten wie Registrierung von Clients oder (im Fall, dass der Teilnehmer Datendienstleister ist) Apps registrieren bzw. anpassen (z.B. Releasewechsel durchführen).

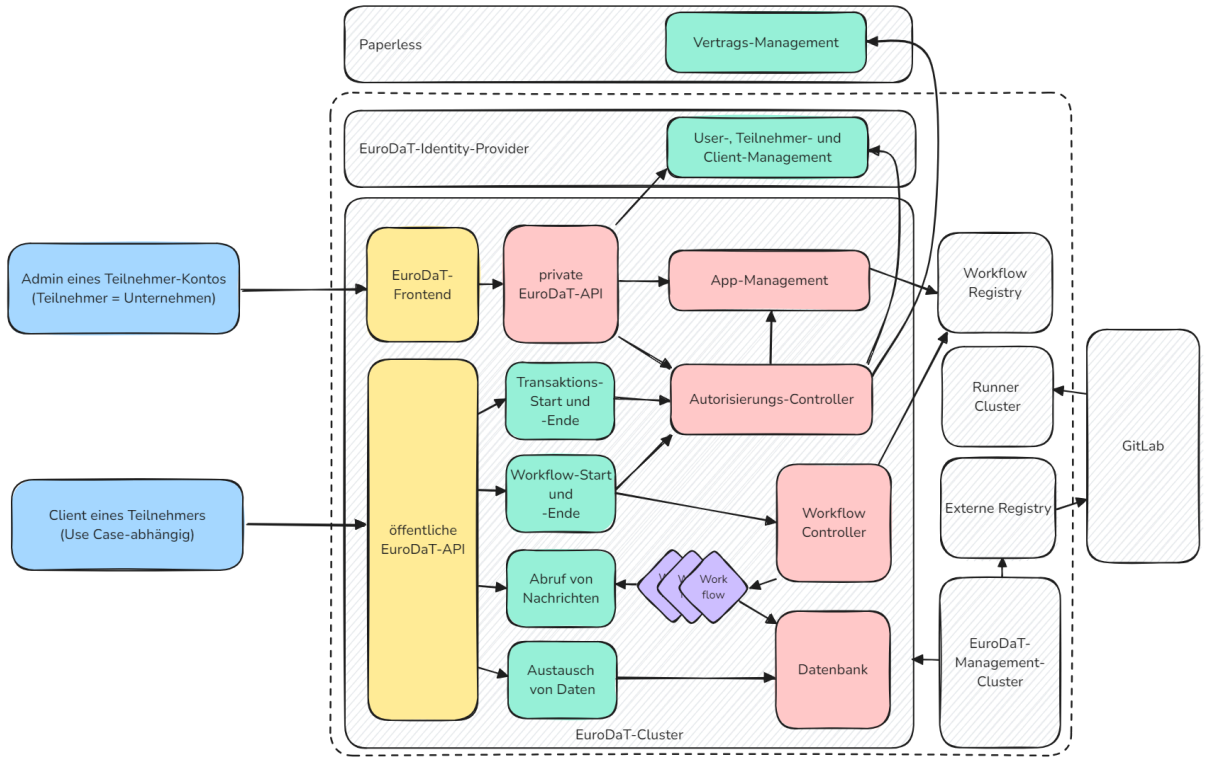
1.2. Architektur der Software

Die Web UI von EuroDaT ist eine Angular-App, die in TypeScript entwickelt wird. Das Vertragsmanagement erfolgt über die angebundene Third Party Software Paperless.

Das Backend der EuroDaT-Plattform ist eine in Kotlin auf Basis des Quarkus-Frameworks entwickelte Micro-Service-Applikation, die in Kubernetes betrieben wird. Alle M2M-Interaktionen (Client-Interaktionen auf Kontroll- und Datenebene sowie Zugriff des Front- auf das Backend) laufen über die EuroDaT REST API (die einen öffentlichen und einen privaten Teil hat).

Authentifizierung und Autorisierung von Usern gegenüber dem Frontend erfolgt über OIDC. Authentifizierung von Clients gegenüber der Public API erfolgt über OIDC Core Authentication Code Flow for Confidential Clients mit private_key_jwt. Als Identity Provider wird ein in der Google Cloud betriebenes Keycloak verwendet.

Die Ausführung von Workflow-Containern wird über Argo Workflows als Workflow Controller orchestriert. Für das Management der Workflow-Container kommt eine dedizierte Artifact Registry zum Einsatz. Im Rahmen von Transaktionen werden logische Datenbanken auf einem PostgreSQL-Server provisioniert. Für das Management der Datenbank-Credentials für die Data Management API (mit Hilfe derer Teilnehmer Daten lesen und schreiben können) wird Hashicorp Vault verwendet. Die anderen Dienste sind selbst entwickelt.



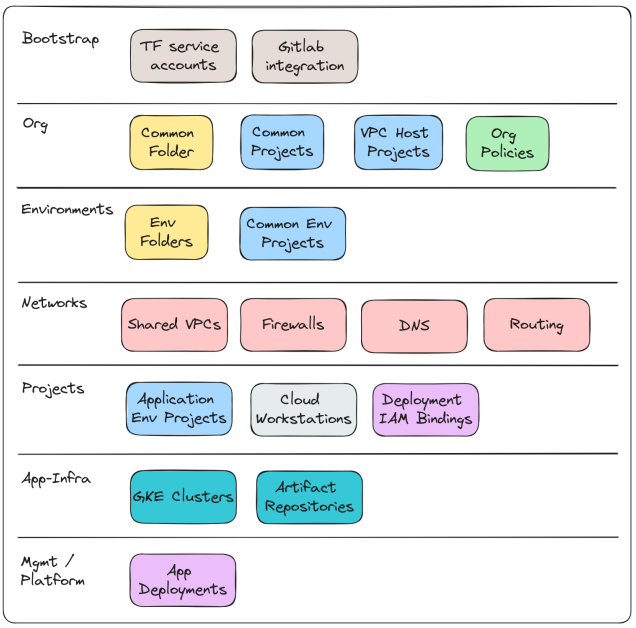
Für das Management und Deployment der EuroDaT-Plattform im EuroDaT-Kubernetes-Cluster nach GitOps-Methodik wird Argo CD eingesetzt, das in einem Management-Kubernetes-Cluster läuft. Die für das Deployment nötigen Images werden über externe Registries bezogen.

Das Deployment verwendet Argo App-CRDs, die auf selbst entwickelte Helm-Charts und pro Stage verwaltete Konfigurationsartefakte verweisen.

Build- und Test-Prozesse werden durch das EuroDaT-eigene GitLab (Cloud) ausgeführt. Dieses verwendet selbst gehostete Runner in einem entsprechenden Kubernetes-Cluster. Der EuroDaT-Quellcode und erzeugte Artefakte werden umfassend qualitätsgesichert. Dazu dienen u.a. Unit, Integration und E2E Tests mit JUnit, GitLab SAST, OWASP Dependency Check, SonarCloud und Trivy. Dependency Updates werden regelmäßig mit Renovate durchgeführt.

1.3. Architektur der Cloud-Organisation

Die Google Cloud-Organisation von EuroDaT wird mit Terraform aus GitLab heraus verwaltet. Ihre Architektur basiert auf dem Google Enterprise Blueprint und ist wie folgt in Repositories organisiert.



Die Organisation ist in Stage-spezifische Folder und einige allgemeine Folder strukturiert. EuroDaT hat die vier Stages Development, QA, Integration und Production. QA ist für auf die EuroDaT-Plattform bezogene Tests. Integration ist für Integrationstests, bei denen auf EuroDaT registrierte Apps zum Einsatz kommen.

Das Networking und das Rollen- und Rechtemanagement sind nach dem Need-to-Know-Prinzip strukturiert. Entwicklung erfolgt ausschließlich über Cloud Workstations. Die Zugriffssteuerung ist identitätsbasiert.

1.4. Quellcode und vorhandene Dokumentation

Der Quellcode der EuroDaT-Plattform ist Open Source und hier einsehbar: https://gitlab.com/eurodat.

Die Dokumentations-Website ist https://docs.eurodat.org. Hier finden sich auch Informationen zum Logging, Monitoring und Tracing.

2. safeAML

safeAML besteht aus einer App, die auf EuroDaT betrieben wird (siehe oben), sowie aus einem dedizierten Client, der von Teilnehmern genutzt werden kann, um mit der App zu interagieren. safeAML wird von EuroDaT selbst weiterentwickelt und betrieben.

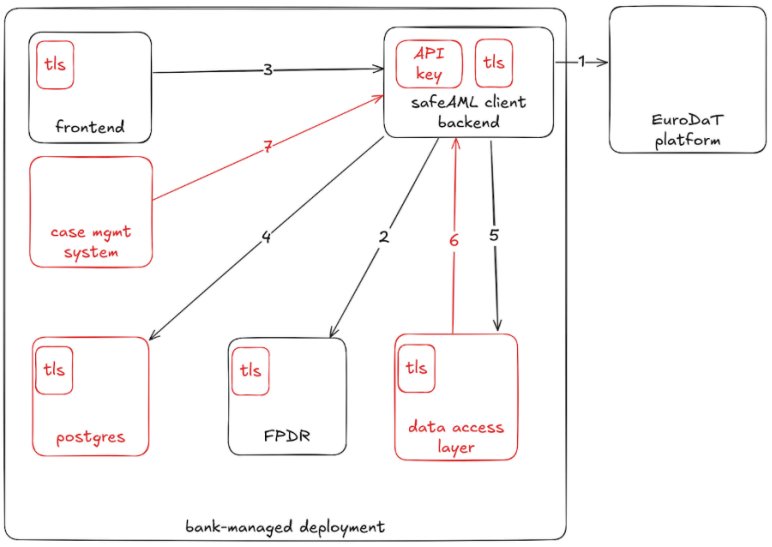
2.1. Allgemeines

Die Teilnehmer, die safeAML nutzen, sind Banken bzw. deren Compliance-Mitarbeiter:innen. Die Grundidee von safeAML ist es, die Nachforschung zu potenziell Geldwäsche-verdächtigen Buchungen zu automatisieren. Compliance-Mitarbeiter:innen einer Bank geben die Buchung ein, sie wird verschlüsselt an EuroDaT übermittelt und die safeAML-App auf EuroDaT sammelt iterativ dazugehörige Buchungen von angeschlossenen Banken ein, um ein Buchungsnetzwerk zur Darstellung des Geldflusses zu erstellen. Dieses wird nach einer Datenminimierung an die anfragende Bank übermittelt.

2.2. Architektur

Der safeAML-Client besteht aus drei-Komponenten:

1) Dem Frontend, einer Vue-App in TypeScript, in dem Compliance-Mitarbeiter ihre Nachforschungs-Prozesse verwalten können.
2) Dem FPDR-Dienst (für “Format-Preserving De-Risking”), der für die Pseudonymisierung bzw. De-Pseudonymisierung der Buchungsdaten zuständig ist. Dies ist eine Python-App auf Basis von FastAPI und verwendet proprietäre Verschlüsselungsalgorithmen.
3) Dem safeAML Client Backend, einer Java-App auf Basis von Spring Boot, die die Orchestrierung der bankseitigen Datenanfragen und die Interaktion mit der EuroDaT-Plattform automatisiert. Es verwendet Hibernate ORM und Spring Security.



Die Client-Komponenten werden in containerisierter Form von EuroDaT den angeschlossenen Banken zur Verfügung gestellt und von diesen betrieben. Wie im Bild rot dargestellt, haben die Client-Komponenten drei Integrationspunkte auf Bankseite: Die Case Management Systeme, die die API des Backend konsumieren, eine PostgreSQL-Datenbank, die die Persistenzschicht zum Management der Nachforschungen darstellt und deren Schema durch den EuroDaT definiert wird, sowie eine Datenzugriffsschicht zum Abrufen von Buchungsdaten, die von der Bank zu verwalten ist, und deren API durch EuroDaT spezifiziert wird.

Die safeAML-App ist eine EuroDaT-App mit einem einzigen Workflow, der gestartet wird, wenn ein Compliance-Mitarbeiter einer angeschlossenen Bank eine neue Nachforschung startet. Dieser Workflow ist eine Python-App, die SQLModel/SQLAlchemy ORM für Zugriff auf EuroDaT-interne Datenendpunkte und httpx für Zugriffe auf EuroDaT-interne API verwendet. Sie orchestriert die Datensammlung über die angeschlossenen Banken und führt die Datenminimierung der Ergebnis-Reports vor Auslieferung durch.

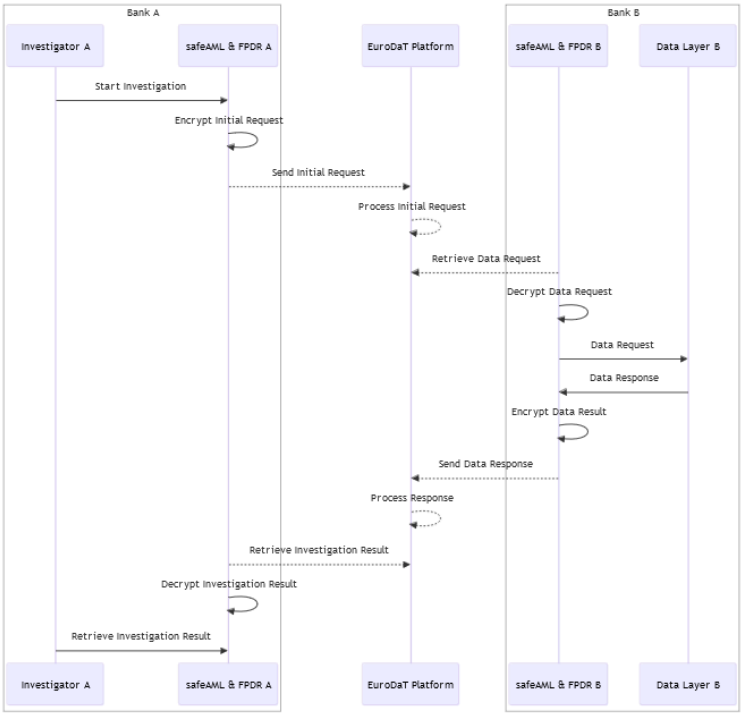
Alle Integrationspunkte zwischen verschiedenen Punkten werden soweit möglich über die Autogenerierung von Clients und Server-Stubs (mit openapi-generator) bzw. die Autogenerierung von SQL-DDL-Statements aus Modellen (mit Flyway für den Client bzw. Alembic für die App) verwaltet.

Die safeAML-App wird durch EuroDaT auf der EuroDaT-Plattform betrieben.

safeAML-Client und -App werden durch EuroDaT in einem dedizierten GitLab (nicht öffentlich) umfassend qualitätsgesichert. Dazu dienen u.a. Unit, Integration und E2E Tests mit JUnit bzw. Pytest, GitLab SAST, SonarCloud und Trivy.

2.3. Beschreibung der Funktionalität

safeAML sammelt Daten mit dem Ziel, die Herkunft von Mitteln zu klären, um Compliance-Mitarbeiter:innen eine Entscheidungsgrundlage für eine ggf. zu erfolgende Geldwäsche-Verdachtsmeldung zu geben. Der grobe Ablauf der Datensammlung durch safeAML ist wie folgt:



Dabei beginnt die Datensammlung bei einer Buchung, die bei der anfragenden Bank A eingegangen ist. Bei der die Buchung sendenden Bank B werden dann dazugehörige Buchungen abgefragt. Iterativ wird bei den Ursprungsbanken nach weiteren Buchungen gefragt, usw., bis Abbruchkriterien erfüllt sind.